

FFFFFFFFFFFFFFFF	DDDDDDDDDDDDDD	LLL
FFFFFFFFFFFFFFFF	DDDDDDDDDDDDDD	LLL
FFFFFFFFFFFFFFFF	DDDDDDDDDDDDDD	LLL
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFFFFFFFFFFFFFF	DDD	DDD
FFFFFFFFFFFFFFF	DDD	DDD
FFFFFFFFFFFFFFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDD	DDD
FFF	DDDDDDDDDDDDDD	LLLLLLLLLLLLLLLL
FFF	DDDDDDDDDDDDDD	LLLLLLLLLLLLLLLL
FFF	DDDDDDDDDDDDDD	LLLLLLLLLLLLLLLL

FFFFFFFF	DDDDDDDD	LL	PPPPPPPP	AAAAAA	RRRRRRRR	DDDDDDDD	EEEEEEEEEE	FFFFFFFF				
FFFFFFFF	DDDDDDDD	LL	PPPPPPPP	AAAAAA	RRRRRRRR	DDDDDDDD	EEEEEEEEEE	FFFFFFFF				
FF	DD	DD	PP	PP	AA	AA	RR	RR	DD	DD	EE	FF
FF	DD	DD	PP	PP	AA	AA	RR	RR	DD	DD	EE	FF
FF	DD	DD	PP	PP	AA	AA	RR	RR	DD	DD	EE	FF
FF	DD	DD	PP	PP	AA	AA	RR	RR	DD	DD	EE	FF
FFFFFFFF	DD	DD	PPPPPPPP	AA	AA	RRRRRRRR	DD	DD	DD	DD	EEEEEEEE	FFFFFFFF
FFFFFFFF	DD	DD	PPPPPPPP	AA	AA	RRRRRRRR	DD	DD	DD	DD	EEEEEEEE	FFFFFFFF
FF	DD	DD	PP		AAAAAAAAAA	RR	RR		DD	DD	EE	FF
FF	DD	DD	PP		AAAAAAAAAA	RR	RR		DD	DD	EE	FF
FF	DD	DD	PP		AA	AA	RR	RR	DD	DD	EE	FF
FF	DD	DD	PP		AA	AA	RR	RR	DD	DD	EE	FF
FF	DD	DD	PP		AA	AA	RR	RR	DD	DD	EE	FF
FF	DDDDDDDD	LLLLLLLLLL	PP		AA	AA	RR	RR	DDDDDDDD	EEEEEEEEEE	FF	FF
FF	DDDDDDDD	LLLLLLLLLL	PP		AA	AA	RR	RR	DDDDDDDD	EEEEEEEEEE	FF	FF

```

SSSSSSSS DDDDDDDD LL
SSSSSSSS DDDDDDDD LL
SS        DD        DD LL
SS        DD        DD LL
SS        DD        DD LL
SS        DD        DD LL
        SSSSSS DD        DD LL
        SSSSSS DD        DD LL
                SS DD        DD LL
                SS DD        DD LL
                SS DD        DD LL
                SS DD        DD LL
SSSSSSSS DDDDDDDD LLLLLLLLLL
SSSSSSSS DDDDDDDD LLLLLLLLLL

```


{
{ Version: 'V04-000'
{

{*****
{*
{* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
{* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
{* ALL RIGHTS RESERVED. *
{*
{* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
{* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
{* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
{* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
{* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
{* TRANSFERRED. *
{*
{* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
{* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
{* CORPORATION. *
{*
{* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
{* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
{*
{*
{*****

```
module $FDLDEF3;
```

```
/**      These fields are found in FDL$AB_CTRL
/*
```

```
aggregate FDLDEF3 union prefix FDL$;
```

```
  FDLDEF BITS0 structure;
```

```
    STATUS bitfield mask length 3;
```

```
    WARNING bitfield mask;
```

```
    PRIMARY bitfield mask;
```

```
    NEWPRI bitfield mask;
```

```
    SECONDARY bitfield mask;
```

```
    INITIAL bitfield mask;
```

```
    COMMENT bitfield mask;
```

```
    LINECMY bitfield mask;
```

```
    PCALL bitfield mask;
```

```
    DCL bitfield mask;
```

```
    STRING SPEC bitfield mask;
```

```
    USED STRING bitfield mask;
```

```
    APOST-PRES bitfield mask;
```

```
    QUOTE-PRES bitfield mask;
```

```
    REPARSE bitfield mask;
```

```
    DFLT PRES bitfield mask;
```

```
    STVACID bitfield mask;
```

```
    GCALL bitfield mask;
```

```
    FULLGEN bitfield mask;
```

```
    DEALLOC bitfield mask;
```

```
/* Status code for processing
```

```
/* A warning message has been issued for this secondary
```

```
/* Primary has been parsed
```

```
/* A new primary has been parsed
```

```
/* Secondary has been parsed
```

```
/* Initial pass
```

```
/* Secondary comment has been detected
```

```
/* Line comment has been detected
```

```
/* This is an EDF Parse call
```

```
/* Called by a DCL utility
```

```
/* An FDL STRING
```

```
/* FDL STRING has been set up
```

```
/* An apostrophe was found by the pre_parse
```

```
/* A quotation mark was found by the pre_parse
```

```
/* Doing a parse into a parse
```

```
/* The DFLT FDL SPEC argument was present
```

```
/* FDL$GSL_STNUMPTR is valid
```

```
/* This is an EDF Generate call
```

```
/* Generate the full FDL spec
```

```
/* Make FDL$$CHECK_BLOCKS deallocate the
```

```
/* RMS control blocks after it checks them
```

```
end FDLDEF_BITS0;
```

```
/*      Codes found in FDL$GSL_PRIMARY
/*
```

```
  constant(
```

```
    DUMMY_PRIMARY$
```

```
    , ACCESS
```

```
    , ACL
```

```
    , ANALA
```

```
    , ANALK
```

```
    , AREA
```

```
    , CONNECT
```

```
    , DATE
```

```
    , FILE
```

```
    , "IDENT"
```

```
    , JNL
```

```
    , KEY
```

```
    , RECORD
```

```
    , SHARING
```

```
    , SYSTEM
```

```
    , TITLE
```

```
/* Dummy_primary$
```

```
/* Access
```

```
/* Access Control Lists
```

```
/* Analysis_of_area
```

```
/* Analysis_of_key
```

```
/* Area
```

```
/* Connect
```

```
/* Date
```

```
/* File
```

```
/* Ident
```

```
/* Journal
```

```
/* Key
```

```
/* Record
```

```
/* Sharing
```

```
/* System
```

```
/* Title
```

```
/*      LAST PRIMARY
```

```
/*
```

```
    PRITAB_SIZE
```

```
    } equals 0 increment 1 prefix FDL tag $C;
```

```
/*      Bits defined for FDL$AB_PRIMCTRL
```



```

/*
  FDLDEF BITS1 structure;
    ACCESS bitfield mask;          /* Access
    ACL bitfield mask;             /* Access Control List
    ANALA bitfield mask;          /* Analysis_of_area
    ANALK bitfield mask;          /* Analysis_of_key
    AREA bitfield mask;           /* Area
    CONNECT bitfield mask;        /* Connect
    DATE bitfield mask;           /* Date
    FILE bitfield mask;           /* File
    "IDENT" bitfield mask;        /* Ident
    JNL bitfield mask;            /* Journal
    KEY bitfield mask;            /* Key
    RECORD bitfield mask;         /* Record
    SHARING bitfield mask;        /* Sharing
    SYSTEM bitfield mask;         /* System
    TITLE bitfield mask;          /* Title
  end FDLDEF_BITS1;

/*      Single field switch for YES - NO qualifiers
/*
  constant FALSE      equals 0 prefix FDL tag $C;  /* No
  constant TRUE       equals -1 prefix FDL tag $C; /* Yes

/*      Secondary codes for each primary
/*      These codes are found in FDL$GL_SECONDARY
/*
/*      Qualifiers for each secondary are listed seperatly
/*
/*      !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
/*      REMEMBER TO UPDATE XXX BEG AND XXX END MARKERS IF AN ATTRIBUTE IS
/*      ADDED/SUBTRACTED ONTO/OFF EITHER END OF A PRIMARY SECTION
/*      !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

/*      Access primary
/*
  constant(
    DUMMY_SECONDARY$          /* Dummy_secondary$
    , FACBIO                  /* Block I/O only
    , FACDEL                  /* Deletes
    , FACGET                  /* Gets
    , FACPUT                  /* Puts
    , FACBRO                  /* Record and Block I/O
    , FACTRN                  /* Truncate
    , FACUPD                  /* Updates

/*      ACL
/*
    , ACE                      /* Entry

/*      Codes for Analysis_of_area primary
/*
    , RECL                      /* Reclaimed_space

```

/* Analysis_of_key primary
/*

. DFIL
. DKC
. DRC
. DREC
. DSPC
. DELE
. DEPTH
. DUPL
. ICOMP
. IFIL
. ISPC
. L1RCNT
. MDL
. MIL
. RANACC
. RANINS
. SEQACC

/* Data_fill
/* Data_key_compression
/* Data_record_compression
/* Data_record_count
/* Data_space_occupied
/* Deletions
/* Depth
/* Duplicates_per_value
/* Index_compression
/* Index_fill
/* Index_space_occupied
/* Level1_record_count
/* Mean_data_length
/* Mean_index_length
/* Random_accesses
/* Random_inserts
/* Sequential_accesses

/* Codes for Area primary
/*

. ALLOC
. BTCONT
. BKT
. CONTG
. EXACT
. EXTND
. POSI
. VOLU

/* Allocation
/* Best_try_contiguous
/* Bucket_size
/* Contiguous
/* Exact_position
/* Extend
/* Position
/* Volume

/* Codes for Connect primary
/*

. ASY
. BIO
. BUCODE
. RCTX
. EOF
. FLOA
. FDEL
. KRF
. KGE
. KGT
. KLIM
. LOCMODE
. REA
. RLK
. ULK
. MBC
. MBF
. NLK
. NXR
. RAH
. RRL
. TMENB
. TMO

/* Asynchronous
/* Block_IO
/* Bucket_code
/* Context
/* End_of_file
/* Fill_buckets
/* Fast_delete
/* Key_of_reference
/* Key_greater_equal
/* Key_greater_than
/* Key_limit
/* Locate_mode
/* Lock_on_read
/* Lock_on_write
/* Manual_unlocking
/* Multiblock_count
/* Multibuffer_count
/* Nolock
/* Nonexistent_record
/* Read_ahead
/* Read_regardless
/* Timeout_enable
/* Timeout_period


```

      . TPT                      /* Truncate on put
      . TTCCO                    /* TT_cancel_control_o
      . TTCVT                    /* TT_upcase_input
      . TIPMT                    /* TT_prompt
      . TIPTA                    /* TT_purge_type_ahead
      . TTRNE                    /* TT_read_noecho
      . TTRNF                    /* TT_read_nofilter
      . UIF                      /* Update_if
      . WAT                      /* Wait_for_record
      . WBH                      /* Write_behind

/*      Codes for Date primary
/*
      . BACKUP                  /* Backup
      . CREAT                   /* Creation
      . EXPR                    /* Expiration
      . REV                     /* Revision

/*      Codes for File primary
/*
      . ALL                     /* Allocation
      . BTC                     /* Best_try_contiguous
      . BKTSIZ                  /* Bucket_size
      . CLUSIZ                  /* Cluster_size
      . FCTX                    /* Context
      . CONT                    /* Contiguous
      . CIF                     /* Create_if
      . DFNAM                   /* Default_name
      . DEFWRT                  /* Deferred_write
      . DOC                     /* Delete_on_close
      . DIR                     /* Directory_entry
      . EODEL                   /* Erase_on_delete
      . EXTEN                   /* Extension
      . GBC                     /* Global_buffer_count
      . MTBLSIZ                 /* MT_block_size
      . MTCP                    /* MT_current_position
      . MTNEF                   /* MT_not_eof
      . MTPRO                   /* MT_protection
      . MTREW                   /* MT_rewind / MT_open_rewind
      . MTRWC                   /* MT_close_rewind
      . MAXRECN                 /* Max_record_number
      . MAXVER                  /* Maximize_version
      . NAME                    /* Name
      . BKTUP                   /* Nobackup
      . NFS                     /* Non_file_structured
      . OFP                     /* Output_file_parse
      . ORG                     /* Organization
      . OWNER                   /* Owner
      . POC                     /* Print_on_close
      . PROT                    /* Protection
      . READC                   /* Read_check
      . REVISN                  /* Revision
      . SQO                     /* Sequential_only
      . SOC                     /* Submit_on_close
      . SUPER                   /* Superscede
      . TEMPO                   /* Temporary
```

```

      , TOC                      /* Truncate_on_close
      , UFO                      /* User_file_open
      , WIN                      /* Window_size
      , WRITEC                   /* Write_check

/*      Codes for Journal primary
/*
      , AFTIM                    /* After_image
      , AFTNAM                   /* After_name
      , AUDIT                    /* Audit_trail
      , AUDNAM                   /* Audit_name
      , BEFIM                    /* Before_image
      , BEFNAM                   /* Before_name
      , RU                       /* Recovery_unit

/*      Codes for Key primary
/*
      , CHANGE                   /* Changes
      , DAREA                    /* Data_area
      , DFILL                    /* Data_fill
      , DATKC                    /* Data_key_compression
      , DATRC                    /* Data_record_compression
      , DUPS                     /* Duplicates
      , IAREA                    /* Index_area
      , IDXC                     /* Index_compression
      , IFILL                    /* Index_fill
      , LAREA                    /* Level1_index_area
      , KYNAME                   /* Name
      , NULL                     /* Null_key
      , NULLVAL                  /* Null_value
      , PROL                     /* Prologue_version
      , SEGLEN                   /* Segment Length
      , SEGPOS                   /* position
      , SEGTYP                   /* type

/*      Codes for Record primary
/*
      , BLKSPN                   /* Block_span
      , CARCTRL                  /* Carriage_control
      , VFCSIZ                   /* Control_field_size
      , FMT                      /* Format
      , SIZE                     /* Record_size

/*      Sharing primary
/*
      , SHRDEL                   /* Deletes
      , SHRGET                   /* Gets
      , SHRMSE                   /* Multi-stream connects
      , SHRNIL                   /* Dissallow sharing
      , SHRPUT                   /* Puts
      , SHRUPD                   /* Updates
      , SHRUPI                   /* User provided interlocking

/*      Codes for System primary
/*
      , DEVICE                   /* Device
```



```
      , SOURCE                      /* Source
      , TARGET                      /* Target

/** THE LAST SECONDARY FOLLOWS:
/*
      , SECTAB_SIZE
      ) equals 0 increment 1 prefix FDL tag $C;

/* The following are markers which are useful to FDL$GENERATE
/*
constant ACCESS_BEG equals FDL$C_FACBIO prefix FDL$ tag C;
constant ACCESS_END equals FDL$C_FACUPD prefix FDL$ tag C;

constant ACL_BEG equals FDL$C_ACE prefix FDL$ tag C;
constant ACL_END equals FDL$C_ACE prefix FDL$ tag C;

constant ANALYSIS_OF_AREA_BEG equals FDL$C_RECL prefix FDL$ tag C;
constant ANALYSIS_OF_AREA_END equals FDL$C_RECL prefix FDL$ tag C;

constant ANALYSIS_OF_KEY_BEG equals FDL$C_DFIL prefix FDL$ tag C;
constant ANALYSIS_OF_KEY_END equals FDL$C_SEQACC prefix FDL$ tag C;

constant AREA_BEG equals FDL$C_ALLOC prefix FDL$ tag C;
constant AREA_END equals FDL$C_VOLU prefix FDL$ tag C;

constant CONNECT_BEG equals FDL$C_ASY prefix FDL$ tag C;
constant CONNECT_END equals FDL$C_WBH prefix FDL$ tag C;

constant DATE_BEG equals FDL$C_BACKUP prefix FDL$ tag C;
constant DATE_END equals FDL$C_REV prefix FDL$ tag C;

constant FILE_BEG equals FDL$C_ALL prefix FDL$ tag C;
constant FILE_END equals FDL$C_WRITEC prefix FDL$ tag C;

constant JOURNAL_BEG equals FDL$C_AFTIM prefix FDL$ tag C;
constant JOURNAL_END equals FDL$C_RU prefix FDL$ tag C;

constant KEY_BEG equals FDL$C_CHANGE prefix FDL$ tag C;
constant KEY_END equals FDL$C_SEGTYP prefix FDL$ tag C;

constant RECORD_BEG equals FDL$C_BLKSPN prefix FDL$ tag C;
constant RECORD_END equals FDL$C_SIZE prefix FDL$ tag C;

constant SHARING_BEG equals FDL$C_SHRDEL prefix FDL$ tag C;
constant SHARING_END equals FDL$C_SHRUP1 prefix FDL$ tag C;

constant SYSTEM_BEG equals FDL$C_DEVICE prefix FDL$ tag C;
constant SYSTEM_END equals FDL$C_TARGET prefix FDL$ tag C;

/* Qualifiers
/*
/* These codes are found in FDL$GL_QUALIFIER
/*

/* Qualifiers for the Area secondary
/*
```

```
constant(
  ANYPOS          /* Any Cylinder
  , CLUSPOS       /* Cluster
  , CYLPOS        /* Cylinder
  , FIDPOS        /* File_ID
  , FNMPOS        /* File_name
  , LOGPOS        /* Logical
  , NOPOS         /* None
  , VIRPOS        /* Virtual

/*
  Qualifiers for the Record secondary
/*
  , NONE          /* None
  , CR            /* Carrage return
  , FTN           /* Fortran
  , PRINT         /* Print

  , UDF           /* UNDEFINED
  , FIX           /* FIXED
  , VAR           /* VARIABLE
  , VFC           /* VFC
  , STM           /* STREAM
  , STMLF         /* STREAM_LF
  , STMCR         /* STREAM_CR

/*
  Qualifiers for the Journal secondary
/*
  , IF_IN         /* If_in_recovery_unit
  , NEC           /* Necessary_to_write
  , NEVER         /* Never_RU_journal

/*
  Qualifiers for the System secondary
/*
  , IAS           /* IAS
  , RSTS          /* RSTS/E
  , M             /* RSX-11M
  , MPLUS         /* RSX-11M-PLUS
  , RT            /* RT-11
  , TRAX          /* TRAX-11
  , VMS           /* VAX/VMS

/*
  Qualifiers for the File secondary
/*
  , SEQ           /* SEQUENTIAL
  , REL           /* RELATIVE
  , IDX           /* INDEXED
  , HSH           /* HASHED

/*
  Qualifiers for the Key secondary
/*
  , STG           /* STRING
  , IN2           /* INT2
  , BN2           /* BIN2
  , IN4           /* INT4
```



```
, BN4
, IN8
, BN8
, PAC
```

```
/* BIN4
/* INT8
/* BIN8
/* DECIMAL
```

```
) equals 0 increment 1 prefix FDL tag $C;
```

```
/* Constants for FDLGENTAB
/*
```

```
constant(
```

```
    FAB
,   RAB
,   XAB
,   NAM
```

```
) equals 0 increment 1 prefix FDL tag $C;
```

```
constant(
```

```
    DUMMY
,   BYTE
,   WORD
,   LONG
,   QUAD
,   OCTA
,   SWITCH
,   STRING
,   QUALIFIER
,   SPECIAL
```

```
) equals 0 increment 1 prefix FDL tag $C;
```

```
/** Parse data stuff
```

```
/**
/** These structures must be contiguous so that EDF can access them from
/** one point. If they need to be changed consult with the owner of edf
/**
```

```
/** FDL$AL_BLOCK:
```

ctrl	0
pcall	1
primary	2
prinum	3
prictrl	4
secondary	5
secnum	6
-----	7

```
/**
/**
/**
/**      secctrl
/**
/**
/**
/**
/**      qualifier      13
/**      number      14
/**      switch      15
/**      owner uic      16
/**      spare1      17
/**      protection      18
/**      fid 1      19
/**      fid 2      20
/**      fid 3      21
/**      - date/time quadword -      22
/**
/**      24
/**      string desc
/**
/**      26
/**      comment desc
/**
/**      28
/**      line desc
/**
/**      30
/**      upcased desc
/**
/**      line count      32
/**      33
/**      item desc
/**
/**      gcall      35 = FDL$K_BLOCK_END
/**
```



```
/**
/**
/** FDL$K_BLOCK_END equals the offset to the last longword in FDL$AL_BLOCK
/**

constant CTRL          equals 0                prefix FDL tag $L;
constant PCALL         equals FDL$$_CTRL + 1    prefix FDL tag $L;
constant PRIMARY      equals FDL$$_PCALL + 1    prefix FDL tag $L;
constant PRINUM       equals FDL$$_PRIMARY + 1  prefix FDL tag $L;
constant PRICTRL      equals FDL$$_PRINUM + 1   prefix FDL tag $L;
constant SECONDARY    equals FDL$$_PRICTRL + 1  prefix FDL tag $L;
constant SECNUM       equals FDL$$_SECONDARY + 1 prefix FDL tag $L;

/** FDL$K_SCTRL_LONG is the number of longwords in FDL$AB_SECCTRL
/** FDL$K_SCTRL_VEC is the number of bits in FDL$AB_SECCTRL
/** Each longword has enough bits to map 32 unique secondary attributes

constant SCTRL_LONG equals 6                prefix FDL tag $K;
constant SCTRL_VEC  equals FDL$K_SCTRL_LONG * 32 prefix FDL tag $K;
constant SECCTRL    equals FDL$$_SECNUM + 1  prefix FDL tag $A;
constant QUALIFIER  equals
FDL$$_SECCTRL + FDL$K_SCTRL_LONG prefix FDL tag $L;

constant NUMBER      equals FDL$$_QUALIFIER + 1 prefix FDL tag $L;
constant SWITCH      equals FDL$$_NUMBER + 1  prefix FDL tag $L;
constant OWNER UIC   equals FDL$$_SWITCH + 1  prefix FDL tag $L;
constant SPARET      equals FDL$$_OWNER UIC + 1 prefix FDL tag $L;
constant PROTECTION  equals FDL$$_SPARET + 1  prefix FDL tag $L;
constant FID1        equals FDL$$_PROTECTION + 1 prefix FDL tag $L;
constant FID2        equals FDL$$_FID1 + 1    prefix FDL tag $L;
constant FID3        equals FDL$$_FID2 + 1    prefix FDL tag $L;
constant DATE TIME   equals FDL$$_FID3 + 1    prefix FDL tag $Q;
constant STRING      equals FDL$$_DATE TIME + 2 prefix FDL tag $Q;
constant COMMENT     equals FDL$$_STRING + 2  prefix FDL tag $Q;
constant LINE        equals FDL$$_COMMENT + 2  prefix FDL tag $Q;
constant UPCASED     equals FDL$$_LINE + 2     prefix FDL tag $Q;
constant STMNTNUM    equals FDL$$_UPCASED + 2  prefix FDL tag $L;
constant ITEM        equals FDL$$_STMNTNUM + 1  prefix FDL tag $Q;
constant GCALL       equals FDL$$_ITEM + 2     prefix FDL tag $L;
constant BLOCK_END   equals FDL$$_GCALL       prefix FDL tag $K;

/*      Misc.
/*
/*      Max size of the fdl line
/*
constant MAXLINE      equals 1024                prefix FDL tag $K;
constant CLEAR        equals 0                  prefix FDL tag $C;

end FDLDEF3;
end_module $FDLDEF3;
```


0176 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY